Assessment #2

Submission deadline: 27th March 2022 11:59pm AEST via Moodle

Instruction

Below are the coding tasks that you need to complete individually for assessment 2. You should download the IntelliJ project folder as below and unzip it. Then work on the tasks in the project folder.

Assessment2.zip

This assessment is worth 6% of the unit total. It contains 60 marks, which has two components.

- Task correctness has marks
  - Task 1 has 17 marks
  - Task 2 has 7 marks
  - Task 3 has 10 marks
  - Task 4 has 9 marks
  - Task 5 has 7 marks
- Code readability & documentation has 5 marks
- Code development has 5 marks

Academic Integrity

Please be reminded of the academic integrity mentioned in Week 01. You should code alone and ask the unit staff for help. Do not post your code in public forums.

Task 1 (W3 - 17 marks)

Code in `task1()` method in the Assessment2 class project. Write and test a boolean expression that returns true if an integer variable "n" is in the range 1 to 100 inclusive but not an even integer in the range 40 to 50 inclusive. Design a set of test values and use these to test that your expression works as expected.

An output example will be " `n : 3 expect true get true` "

Task 2 (W3 - 7 marks)

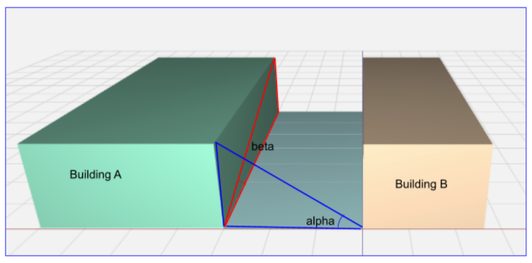
Code in `task2()` method in the Assessment2 class project. The assignment operator allows you to assign the value of one variable to another variable but how can you swap the values in two variables?
Declare two boolean variables with initial values. Write additional code to swap these values. You must do this using only variables. Print using `System.out.println()` before and after swap.

Hint: you can use additional variable.

Task 3 (W3 - 10 marks)

An engineer was asked to estimate some measurements for the client who wants to know an off-the-cuff estimate on how many stone slabs he would need to fill the land area between two of the buildings that he owns. The dimension of the stone slabs are 1m by 1m and can be cut down if required. The engineer was able to use an app on his smartphone (that utilises the sensors) to visually estimate some angles between the buildings as below.

- Angle alpha between the base of building B to the top of building A as 53.13°
- Angle beta from the base of building A to the top (on the opposite side) of building A as being 41.00°
- The height of the buildings are reported as 20.00m



Code in `task3()` method to solve the problem and find out how many stone slabs are required (rounded to the next largest whole number). You only have to consider the area to be filled.

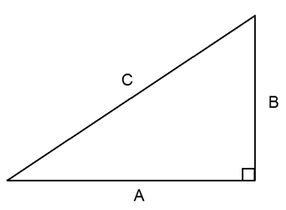
Hint: Note that to get the distance between Building A & B and the length of building A, you need to apply the Law of Tangents. You will need `Math.PI`, `Math.tan()` and `Math.ceil()` from Java Math library. Note that the angle given is in degree and you need to convert to radians.

Task 4 (W3 - 9 marks)

Code in `task4()` method in the Assessment2 class project. Declare 2 int variables x and y and input their values using Scanner Class. Compute the result for both integers using "bitwise and (&)" and "bitwise exclusive OR (^)" operators and print the output individually using `System.out.println()`. Explain step by step how the results are obtained as inline comment.

Task 5 (W4 - 7 marks)

Pythagoras's theorem is the square of the length of the hypotenuse of a right triangle equals the sum of the squares of the lengths of the other two sides. Code a method called **pythagorasTheorem** that returns the result as below equation.



$$c = \sqrt{a^2 + b^2}$$

- It takes in two double datatype **(FORMAL)** parameters, **a** and **b**.
- It uses at least one local variable to calculate the result, **c** using the equation above.
- It prints the result within the method using `System.out.println()`
- It **returns** the result appropriately.

Hint: You will need `Math.sqrt()` from Java Math library.

Once done, test your method in the main class to ensure it is working as expected.

Code Readability (5 marks)

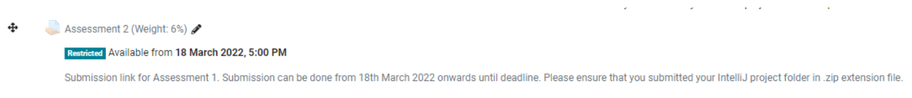
Overall code submission must be well organised and very easy to follow included but not limited to code indentation, code consistency, effective use of whitespace etc.

Code Development & Documentation (5 marks)

Overall code submission demonstrates correct syntax usage and meaningful naming conventions.Code documentations/inline comments are thorough and in detail.

Submission Instruction

Please submit your IntelliJ project folder as a .zip file and submit to via Moodle as below. If you are not sure how to zip your project, please refer to the video here.



Marking Rubric

Assessment#2 Marking Rubric.pdf